METHOD AND SYSTEM FOR ADJUSTING THE TIMING OFFSET BETWEEN A CLOCK SIGNAL AND RESPECTIVE DIGITAL SIGNALS TRANSMITTED ALONG WITH THAT CLOCK SIGNAL, AND MEMORY DEVICE AND COMPUTER SYSTEM USING SAME

TECHNICAL FIELD

The present invention relates generally to semiconductor memories and other integrated circuit devices, and is directed, more particularly, to synchronizing digital signals being transferred over buses interconnecting such devices.

BACKGROUND OF THE INVENTION

Conventional computer systems include a processor (not shown) coupled to a variety of memory devices, including read-only memories ("ROMs") which traditionally store instructions for the processor, and a system memory to which the processor may write data and from which the processor may read data. The processor may also communicate with an external cache memory, which is generally a static random access memory ("SRAM"). The processor also communicates with input devices, output devices, and data storage devices.

Processors generally operate at a relatively high speed. Processors such as the Pentium III® and Pentium 4® microprocessors are currently available that operate at clock speeds of at least 400 MHz. However, the remaining components of existing computer systems, with the exception of SRAM cache, are not capable of operating at the speed of the processor. For this reason, the system memory devices, as well as the input devices, output devices, and data storage devices, are not coupled directly to the processor bus. Instead, the system memory devices are generally coupled to the processor bus through a memory controller, bus bridge or similar device, and the input devices, output devices, and data storage devices are coupled to the processor bus through a bus bridge. The memory controller allows the system memory devices to operate at a lower clock

frequency that is substantially lower than the clock frequency of the processor. Similarly, the bus bridge allows the input devices, output devices, and data storage devices to operate at a substantially lower frequency. Currently, for example, a processor having a 1 GHz clock frequency may be mounted on a mother board having a 133 MHz clock frequency for controlling the system memory devices and other components.

Access to system memory is a frequent operation for the processor. The time required for the processor, operating, for example, at 1 GHz, to read data from or write data to a system memory device operating at, for example, 133 MHz, greatly slows the rate at which the processor is able to accomplish its operations. Thus, much effort has been devoted to increasing the operating speed of system memory devices.

System memory devices are generally dynamic random access memories ("DRAMs"). Initially, DRAMs were asynchronous and thus did not operate at even the clock speed of the motherboard. In fact, access to asynchronous DRAMs often required that wait states be generated to halt the processor until the DRAM had completed a memory transfer. However, the operating speed of asynchronous DRAMs was successfully increased through such innovations as burst and page mode DRAMs which did not require that an address be provided to the DRAM for each memory access. More recently, synchronous dynamic random access memories ("SDRAMs") have been developed to allow the pipelined transfer of data at the clock speed of the motherboard. However, even SDRAMs are incapable of operating at the clock speed of currently available processors. Thus, SDRAMs cannot be connected directly to the processor bus, but instead must interface with the processor bus through a memory controller, bus bridge, or similar device. The disparity between the operating speed of the processor and the operating speed of SDRAMs continues to limit the speed at which processors may complete operations requiring access to system memory.

A solution to this operating speed disparity has been proposed in the form of a computer architecture known as a synchronous link architecture. In the synchronous link architecture, the system memory may be coupled to the processor either directly through the

processor bus or through a memory controller. Rather than requiring that separate address and control signals be provided to the system memory, synchronous link memory devices receive command packets that include both control and address information. The synchronous link memory device then outputs or receives data on a data bus that may be coupled directly to the data bus portion of the processor bus.

An example of a computer system 10 using the synchronous link architecture is shown in Figure 1. The computer system 10 includes a processor 12 having a processor bus 14 coupled through a memory controller 18 and system memory bus 23 to three packetized or synchronous link dynamic random access memory ("SLDRAM") devices 16a-c. The computer system 10 also includes one or more input devices 20, such as a keypad or a mouse, coupled to the processor 12 through a bus bridge 22 and an expansion bus 24, such as an industry standard architecture ("ISA") bus or a peripheral component interconnect ("PCI") bus. The input devices 20 allow an operator or an electronic device to input data to the computer system 10. One or more output devices 30 are coupled to the processor 12 to display or otherwise output data generated by the processor 12. The output devices 30 are coupled to the processor 12 through the expansion bus 24, bus bridge 22 and processor bus 14. Examples of output devices 24 include printers and a video display units. One or more data storage devices 38 are coupled to the processor 12 through the processor bus 14, bus bridge 22, and expansion bus 24 to store data in or retrieve data from storage media (not shown). Examples of storage devices 38 and storage media include fixed disk drives floppy disk drives, tape cassettes and compact-disk read-only memory drives.

In operation, the processor 12 sends a data transfer command via the processor bus 14 to the memory controller 18, which, in turn, communicates with the memory devices 16a-c via the system memory bus 23 by sending the memory devices 16a-c command packets that contain both control and address information. Data is coupled between the memory controller 18 and the memory devices 16a-c through a data bus portion of the system memory bus 23. During a read operation, data is transferred from the

SLDRAMs 16a-c over the memory bus 23 to the memory controller 18 which, in turn, transfers the data over the processor 14 to the processor 12. The processor 12 transfers write data over the processor bus 14 to the memory controller 18 which, in turn, transfers the write data over the system memory bus 23 to the SLDRAMs 16a-c. Although all the memory devices 16a-c are coupled to the same conductors of the system memory bus 23, only one memory device 16a-c at a time reads or writes data, thus avoiding bus contention on the memory bus 23. Bus contention is avoided by each of the memory devices 16a-c on the system memory 22 having a unique identifier, and the command packet contains an identifying code that selects only one of these components.

The computer system 10 also includes a number of other components and signal lines that have been omitted from Figure 1 in the interests of brevity. For example, as explained below, the memory devices 16a-c also receive a master clock signal to provide internal timing signals, a data clock signal clocking data into and out of the memory device 16, and a FLAG signal signifying the start of a command packet.

A typical command packet CA<0:39> for an SLDRAM is shown in Figure 2 and is formed by 4 packet words CA<0:9>, each of which contains 10 bits of data. As will be explained in more detail below, each packet word CA<0:9> is applied on a command-address bus CA including 10 lines CA0-CA9. In Figure 2, the four packet words CA<0:9> comprising a command packet CA<0:39> are designated PW1-PW4. The first packet word PW_1 contains 7 bits of data identifying the packetized DRAM 16a-c that is the intended recipient of the command packet. As explained below, each of the packetized DRAMs is provided with a unique ID code that is compared to the 7 ID bits in the first packet word PW_1. Thus, although all of the packetized DRAMs 16a-c will receive the command packet, only the packetized DRAM 16a-c having an ID code that matches the 7 ID bits of the first packet word PW_1 will respond to the command packet.

The remaining 3 bits of the first packet word PW_1 as well as 3 bits of the second packet word PW_2 comprise a 6 bit command. Typical commands are read and write in a variety of modes, such as accesses to pages or banks of memory cells. The remaining 7

bits of the second packet word PW_2 and portions of the third and fourth packet words PW_3 and PW_4 comprise a 20 bit address specifying a bank, row and column address for a memory transfer or the start of a multiple bit memory transfer. In one embodiment, the 20-bit address is divided into 3 bits of bank address, 10 bits of row address, and 7 bits of column address. Although the command packet shown in Figure 2 is composed of 4 packet words PW1-PW4 each containing up to 10 bits, it will be understood that a command packet may contain a lesser or greater number of packet words, and each packet word may contain a lesser or greater number of bits.

The memory device 16a is shown in block diagram form in Figure 3. Each of the memory devices 16a-c includes a clock generator circuit 40 that receives a command clock signal CCLK and generates a large number of other clock and timing signals to control the timing of various operations in the memory device 16a. The memory device 16a also includes a command buffer 46 and an address capture circuit 48 which receive an internal clock signal ICLK, a command packet CA<0:9> on a 10 bit command-address bus CA, and a terminal 52 receiving a FLAG signal. A memory controller (not shown) or other device normally transmits the command packet CA<0:9> to the memory device 16a in synchronism with the command clock signal CCLK. As explained above, the command packet CA<0:39>, which generally includes four 10-bit packet words PW1-PW4, contains control and address information for each memory transfer. The FLAG signal identifies the start of a command packet, and also signals the start of an initialization sequence. The command buffer 46 receives the command packet from the command-address bus CA, and compares at least a portion of the command packet to identifying data from an ID register 56 to determine if the command packet is directed to the memory device 16a or some other memory device 16b, c. If the command buffer 46 determines that the command is directed to the memory device 16a, it then provides the command to a command decoder and sequencer 60. The command decoder and sequencer 60 generates a large number of internal control signals to control the operation of the memory device 16a during a memory transfer.

The address capture circuit 48 also receives the command packet from the command-address bus CA and outputs a 20-bit address corresponding to the address information in the command packet. The address is provided to an address sequencer 64, which generates a corresponding 3-bit bank address on bus 66, a 10-bit row address on bus 68, and a 7-bit column address on bus 70. The row and column addresses are processed by row and column address paths, as will be described in more detail below.

One of the problems of conventional DRAMs is their relatively low speed resulting from the time required to precharge and equilibrate circuitry in the DRAM array. The SLDRAM 16a shown in Figure 3 largely avoids this problem by using a plurality of memory banks 80, in this case eight memory banks 80a-h. After a read from one bank 80a, the bank 80a can be precharged while the remaining banks 80b-h are being accessed. Each of the memory banks 80a-h receives a row address from a respective row latch/decoder/driver 82a-h. All of the row latch/decoder/drivers 82a-h receive the same row address from a predecoder 84 which, in turn, receives a row address from either a row address register 86 or a refresh counter 88 as determined by a multiplexer 90. However, only one of the row latch/decoder/drivers 82a-h is active at any one time as determined by bank control logic 94 as a function of a bank address from a bank address register 96.

The column address on bus 70 is applied to a column latch/decoder 100, which supplies I/O gating signals to an I/O gating circuit 102. The I/O gating circuit 102 interfaces with columns of the memory banks 80a-h through sense amplifiers 104. Data is coupled to or from the memory banks 80a-h through the sense amps 104 and I/O gating circuit 102 to a data path subsystem 108 which includes a read data path 110 and a write data path 112. The read data path 110 includes a read latch 120 that stores data from the I/O gating circuit 102.

In the memory device 16a shown in Figure 3, 64 bits of data are stored in the read latch 120. The read latch then provides four 16-bit data words to an output multiplexer 122 that sequentially supplies each of the 16-bit data words to a read FIFO buffer 124. Successive 16-bit data words are clocked through the read FIFO buffer 124 in

response to a clock signal RCLK generated by the clock generator 40. The FIFO buffer 124 sequentially applies the 16-bit data words to a driver circuit 128 which, in turn, applies the 16-bit data words to a data bus DQ forming part of the processor bus 14 (see Figure 1). The FIFO buffer 124 also applies two data clock signals DCLK0 and DCLK1 to the driver circuit 128 which, in turn, applies the data clock signals DCLK0 and DCLK1 on respective data clock lines 132 and 133. The data clocks DCLK0 and DCLK1 enable a device, such as the memory controller 18, reading data on the data bus DQ to be synchronized with the data. Particular bits in the command portion of the command packet CA0-CA9 determine which of the two data clocks DCLK0 and DCLK1 is applied by the driver circuit 128. It should be noted that the data clocks DCLK0 and DCLK1 are differential clock signals, each including true and complementary signals, but for ease of explanation, only one signal for each clock is illustrated and described.

The write data path 112 includes a receiver buffer 140 coupled to the data bus 130. The receiver buffer 140 sequentially applies 16-bit data words from the data bus DQ to four input registers 142, each of which is selectively enabled by a signal from a clock generator circuit 144. The clock generator circuit 144 generates these enable signals responsive to the selected one of the data clock signals DCLK0 and DCLK1. The memory controller or processor determines which data clock DCLK0 or DCLK1 will be utilized during a write operation using the command portion of a command packet applied to the memory device 16a. As with the command clock signal CCLK and command packet, the memory controller or other device (not shown) normally transmits the data to the memory device 16a in synchronism with the selected one of the data clock signals DCLK0 and DCLK1. The clock generator 144 is programmed during initialization to adjust the timing of the clock signal applied to the input registers 142 relative to the selected one of the data clock signals DCLK0 and DCLK1 so that the input registers 142 can capture the write data at the proper times. In response to the selected data clock DCLK0 or DCLK1, the input registers 142 sequentially store four 16-bit data words and combine them into one 64-bit data word applied to a write FIFO buffer 148. The write FIFO buffer 148 is clocked by a

signal from the clock generator 144 and an internal write clock WCLK to sequentially apply 64-bit write data to a write latch and driver 150. The write latch and driver 150 applies the 64-bit write data to one of the memory banks 80a-h through the I/O gating circuit 102 and the sense amplifiers 104.

As mentioned above, an important goal of the synchronous link architecture is to allow data transfer between a processor or memory controller and a memory device to occur at a significantly faster rate. However, as the rate of data transfer increases, it becomes more difficult to maintain synchronization of signals transmitted between the memory controller 18 and the memory device 16a. For example, as mentioned above, the command packet CA<0:39> is normally transmitted from the memory controller 18 to the memory device 16a in synchronism with the command clock signal CCLK, and the read and write data are normally transferred between the memory controller 18 and the memory device 16a in synchronism with the selected one of the data clock signals DCLK0 and DCLK1. However, because of unequal signal delays and other factors, the command packet CA<0:39> may not arrive at the memory device 16a in synchronism with the command clock signal CCLK, and write and read data may not arrive at the memory device 16a and memory controller 18, respectively, in synchronism with the selected one of the data clock signals DCLK0 and DCLK1. Moreover, even if these signals are actually coupled to the memory device 16a and memory controller 18 in synchronism with each other, they may loose synchronism once they are coupled to circuits within these respective devices. For example, internal signals require time to propagate to various circuitry in the memory device 16a, differences in the lengths of signal routes can cause differences in the times at which signals reach the circuitry, and differences in capacitive loading of signal lines can also cause differences in the times at which signals reach the circuitry. These differences in arrival times can become significant at high speeds of operation and eventually limit the operating speed of the memory devices 16a and memory controller 18.

The problems associated with varying arrival times are exacerbated as timing tolerances become more restricted with higher data transfer rates. For example, if

the internal clock ICLK derived from the command clock CCLK does not latch each of the packet words CA<0:9> comprising a command packet CA<0:39> at the proper time, errors in the operation of the memory device may result. Similarly, data errors may result during write operations if internal signals developed responsive to the data clocks DCLK0 and DCLK1 do not latch data applied on the data bus DQ at the proper time. During read operations, data errors may likewise result if internal signals in the memory controller 18 developed responsive to the data clock signals DCLK0 and DCLK1 from the memory device 16a do not latch read data applied on the data bus DQ at the proper time. Moreover, even if these respective clocks are initially synchronized, this synchronism may be lost over time during normal operation of the memory device 16a. Loss in synchronism may result from a variety of factors, including temperature variations in the environment in which the memory device 16a is operating, variations in the supply voltage applied to the memory device, and drift in operating parameters of components within the memory device.

One skilled in the art will understand that synchronization of the clock signals CCLK, DCLK0, and DCLK1 is being used to mean the adjusting of the timing of respective internal clock signals derived from these respective external clock signals so the internal clock signals can be used to latch corresponding digital signals at the proper times. For example, the command clock signal CCLK is synchronized when the timing of the internal clock signal ICLK relative to the command clock signal CCLK causes packet words CA<0:9> to be latched at the proper times.

To synchronize the command clock signals CCLK and the data clock signals DCLK0 and DCLK1 during write data operations, the memory controller 18 applies a test bit pattern and (Figure 1) places the memory device 16a in a command and write data synchronization mode. During the synchronization mode, synchronization circuitry within the memory device 16a (not shown in Figure 3) detects the applied bit pattern, places the device in the synchronization mode, and thereafter generates the necessary control signals to control components within the memory device to synchronize the clock signals CCLK, DCLK0, and DCLK1 from the controller 18. The data clock signals DCLK0 and DCLK1

must similarly be synchronized for read operations between the memory controller 18 and memory device 16a.

As mentioned above, an important goal of the synchronous link architecture is to allow data transfer between a processor and a memory device to occur at a significantly faster rate. It should be noted that the phrase "data transfer" as used herein includes all digital signals transferred to and from the memory device 16a, and thus includes signals on the CA and DQ busses as well as the FLAG signal. As the data transfer rate increases, it becomes more difficult to maintain the required timing between signals transmitted between the memory device 16a and the memory controller 18. For example, as mentioned above, the command packet CA<0:39> is normally transmitted to the memory device 16a in synchronization with the command clock signal CCLK, and the data is normally transmitted to the memory device 16a in synchronization with the selected one of the data clock signals DCLK0 and DCLK1. However, because of unequal signal delays and other factors, the command packet words CA<0:9> may not arrive at the memory device 16a in synchronization with the command clock signal CCLK, and the data packet words may not arrive at the memory device 16a in synchronization with the selected data clock signal DCLK0 or DCLK1. Moreover, even if these signals are actually coupled to the memory device 16a in synchronization with each other, this timing may be lost once they are coupled to circuits within the memory device. For example, internal signals require time to propagate to various circuitry in the memory device 16a, differences in the lengths of signal routes can cause differences in the times at which signals reach the circuitry, and differences in capacitive loading of signal lines can also cause differences in the times at which signals reach the circuitry. These differences in arrival times can become significant at high data transfer rates and eventually limit the operating speed of the packetized memory devices.

The problems associated with varying arrival times are exacerbated as timing tolerances become more restricted at higher data transfer rates. For example, if the internal clock ICLK derived from the command clock CCLK does not cause each of the

packet words CA<0:9> comprising a command packet CA<0:39> to latch at the proper time, errors in the operation of the memory device may result. Thus, the timing or phase shift of the internal clock signal ICLK relative to the command clock signal CCLK must be adjusted such that the ICLK signal may be utilized to successfully latch each of the respective command signals CA<0>-CA<9> comprising a packet word CA<0:9>. This is true notwithstanding the varying arrival times of the respective command signals CA<0>-CA<9> within each packet word CA<0:9> relative to the ICLK signal.

Thus, for each of the clock signals CCLK, DCLK0, and DCLK1 the phase shift of respective internal clock signals derived from these respective external clock signals must be adjusted so the internal clock signals can be used to latch corresponding packet words at optimum times. For example, the phase shift of the internal clock signal ICLK relative to the command clock signal CCLK must be adjusted so that all command signals CA<0>-CA<9> in each packet word CA<0:9> are latched at the optimum time.

As the data transfer rate increases, the duration for which each signal CA<0>-CA<9> in a packet word CA<0:9> is valid decreases by a corresponding amount, as will be understood by one skilled in the art. More specifically, the data window or "eye" DE for each of the DQ<0>-DQ<15> signals decreases at higher data transfer rates. As understood by one skilled in the art, the data eye DE for each of the DQ<0>-DQ<9> signals defines the actual duration that each signal is valid after timing skew of the signal is considered. The timing skew of the DQ<0>-DQ<9> signals arises from a variety of timing errors such as loading on the lines of the DQ bus and the physical lengths of such lines. Figure 4 is a timing diagram illustrating the data eyes DE for a number of the DQ<0>-DQ<9> signals. The solid lines indicate the ideal DQ<0>, DQ<1>, and DQ<9> signals, and the dashed lines indicate the worst case potential time skew for each of these signals. The data eyes DE of the DQ<0>, DQ<1>, and DQ<9> signals are defined by time intervals t_0-t_3, t_1-t_4, and t_5-t_7, respectively.

As data eyes DE of the applied signals DQ<0>-DQ<9> decrease at high data transfer rates, it is possible that one or more of these signals in each data packet word

DQ<0:15> will have arrival times such that not all signals in a packet word are simultaneously valid at the memory device 16a, and thus cannot be successfully captured by the internal clock signal ICLK. For example, in Figure 4, the data eye DE of the DQ<0> signal from times t_0-t_3 does not overlap the data eye of the DQ<15> signal from times t_5-t_7. In this situation, the signals DQ<0> and DQ<15> are not both valid at the memory device 16a at the same time so the packet word DQ<0:15> cannot be successfully captured responsive to the RCLK signal. The transition of the RCLK signal at time t_2 could successfully capture the DQ<0> and DQ<1> signals, but not the DQ<15> signal, and, conversely, the transition of the RCLK signal at time t_6 could successfully capture the DQ<15> signal but not the DQ<0> and DQ<1> signals, both of which have already gone invalid at time t_6.

There is a need for synchronizing respective data clock signals and corresponding data packet signals during the transfer of read data between packetized memory devices and a memory controller. Although the foregoing discussion is directed to synchronizing clock signals in packetized memory devices like SLDRAMs, similar problems exist in other types of integrated circuits as well, including other types of memory devices.

SUMMARY OF THE INVENTION

According to one aspect of the present invention, a method adaptively adjusts respective timing offsets of a plurality of digital signals relative to a clock signal being output along with the digital signals to enable a circuit receiving the digital signals successfully to each of the digital signals responsive to the clock signal. The method includes storing in a respective storage circuit associated with each digital signal a corresponding phase command. The phase command defines a particular timing offset between the corresponding digital signal and the clock signal. The clock signal is output along with each digital signal having the timing offset defined by the corresponding phase command. The digital signals are captured responsive to the clock signal and evaluated to

determine if each digital signal was successfully captured. A phase adjustment command is generated to adjust the value of each phase command. The operations of outputting the clock signal through generating a phase adjustment command are repeated for a plurality of phase adjustment commands for each digital signal. A phase command that causes the digital signal to be successfully captured is then selected, and the selected phase command is stored in the storage circuit associated with the digital signal.

According to another aspect of the present invention, a read synchronization circuit adaptively adjusts respective timing offsets of a plurality of digital signals applied on respective signal terminals and an external data clock signal to enable an external device to latch the digital signals responsive to the external data clock signal. The read synchronization circuit includes a plurality of latch circuits, each latch circuit including an input, an output coupled to a respective signal terminal, and a clock terminal. Each latch circuit stores a signal applied on the input and providing the stored signal on the signal terminal responsive to a clock signal applied on the clock terminal. A plurality of phase command registers store phase commands with each register being associated with at least one of the latch circuits.

A clock generation circuit is coupled to latch circuits and the phase command registers and generates a plurality of internal clock signals and the external data clock signal responsive to a read clock signal. Each internal clock signal and the external clock signal has a respective phase shift relative to the read clock signal. The clock generation circuit selects one of the internal clock signals for each latch circuit in response to the associated phase command and applies the selected internal clock signal to the clock terminal of the latch circuit to place digital signals on the corresponding signal terminal with a timing offset determined by the phase shift of the selected internal clock signal.

A control circuit is coupled to the clock generation circuit and the phase command registers and operates in response to a synchronization command to apply synchronization digital signals on the inputs of the latch circuits and to adjust the respective timing offsets between the external data clock signal and the synchronization digital signals

output by each latch circuit by adjusting the respective values of the phase commands. The circuit stores final phase commands in each phase command register that allow the synchronization digital signals to be successfully captured responsive to the external data clock signal. The read synchronization circuit may be utilized in a variety of different types of integrated circuits, including packetized memory devices such as SLDRAMs, nonpacketized devices such as double-data-rate synchronous dynamic random access memories (DDR SDRAMs), and alternative memory architectures having alternative clocking topologies.

BRIEF DESCRIPTION OF THE DRAWINGS

Figure 1 is a block diagram of a conventional computer system including a plurality of packetized memory devices.

Figure 2 is diagram showing a typical command packet received by the packetized memory devices of Figure 1.

Figure 3 is a block diagram of a conventional packetized memory device in the computer system of Figure 1.

Figure 4 is a timing diagram illustrating the effect of timing skews on capturing respective data signals on the data bus at high data transfer rates.

Figure 5 is block diagram of a read synchronization system for a packetized memory device according to one embodiment of the present invention.

Figure 6 is a timing diagram illustrating the operation of the system of Figure 5 in adjusting the respective timing offsets of several data bits relative to a data clock signal to allow the memory controller (Figure 5) to successfully capture all data bits in response to the data clock signal.

Figure 7 is a functional block diagram illustrating a memory system including components of the read synchronization system of Figure 5 according to another embodiment of the present invention.

DETAILED DESCRIPTION OF THE INVENTION

Figure 5 is a functional block diagram of a read synchronization system 400 including a memory controller 402 and packetized memory device 404 according to one embodiment of the present invention. The memory controller 402 applies command packets CA<0:39> and FLAG bits to the memory device 404 over the command-address bus CA and FLAG line, respectively, and transfers data packet words D<0:15> on the data bus DQ. During read operations, the memory controller 402 clocks the received data packet words D<0:15> into the controller responsive to one of the data clocks DCLK0 and DCLK1. Prior to performing read operations, the memory controller 402 places the memory device 404 in a read synchronization mode of operation and adjusts the timing offset of respective bits DQ<0>-DQ<15> in the data packet words D<0:15> relative to the data clock signals DCLK0 and DCLK1 to synchronize the data clock signals for use during normal read operations, as will be explained in more detail below.

One skilled in the art will understand that synchronization of the data clock signals DCLK0 and DCLK1 as described herein means the adjustment of the timing offset of respective bits D<0>-D<15> in the data packet words D<0:15> relative to the data clock signals DCLK0 and DCLK1 such that the memory controller 402 can successfully capture the data packet words D<0:15> responsive to the data clock signals DCLK0 and DCLK1, as will be discussed in more detail below. In the following description, certain details are set forth to provide a sufficient understanding of the present invention. However, it will be clear to one skilled in the art that the invention may be practiced without these particular details. In other instances, well-known circuits, control signals, timing protocols, and software operations have not been shown in detail in order to avoid unnecessarily obscuring the invention.

In the memory device 404, the command buffer and address capture circuit 46 (Figure 3) latches packet words CA<0:9> and FLAG bits applied on the respective command-address bus CA and FLAG line as previously described with reference to Figure 3. A command decoder and sequencer 408 receives the latched command packet

CA<0:39> and FLAG bits from the command buffer and address capture circuit 46 and generates a plurality of control signals 410 to control the operation of various components within the memory device 404 in response to the latched command packet CA<0:39> and FLAG signals. During the read synchronization mode of operation, the command decoder and sequencer 408 generates a phase adjust command word PHADJCMD<0:Y> in response to a phase adjustment command that is applied on the command-address bus CA and latched by the command buffer and address capture circuit 46, as will be described in more detail below.

An up/down phase counter-controller 416 latches the phase adjust command word PHADJCMD from the command decoder and sequencer 408 and operates in response to the latched phase adjust command word to develop a phase command word CMDPH<0:3> and to store the developed phase command word in one of a plurality of phase command registers 434A-P, as will be explained in more detail below. The latched CMDPH<0:3> word stored in the registers 434A-P are designated as phase command words CMDPH0<0:3>-CMDPH15<0:3>, respectively. A plurality of multiplexers 436A-P receive the CMDPH0<0:3>-CMDPH15<0:3> words, respectively, stored in the registers 434A-P, and further receive a plurality of clock signals 438A-N from a delay-locked loop circuit 418 on respective inputs. The delay-locked loop circuit 418 develops the plurality of clock signals 438A-N in response to the RCLK signal, with the clock signals 438A-N having phase shifts, designated ϕ_1-ϕ_N, respectively, relative to the RCLK signal. In the embodiment of Figure 6, the delay-locked loop circuit 418 develops sixteen clock signals 434A-N and maintains a phase shift of 180° between the clock signals 438A and 438N. A more detailed description of one embodiment of a programmable-delay clock generation circuit that may be used as the delay locked loop circuit 418 is described in U.S. Patent Application No. 08/811,918 to Manning, which is incorporated herein by reference.

Each multiplexer 436A-P provides one of the applied clock signals 438A-N on an output in response to the phase command word CMDPH0<0:3>-CMDPH15<0:3> applied from the corresponding register 434A-P. The clock signals output by the

multiplexers 436A-P are designated data read clocks DRCLK0-DRCLK15, respectively, and are applied to clock respective read synchronization data bits RSDW<0>-RSDW<15> into a plurality of data latches 440A-P. A read data pattern generator 424 generates the RSDW<0>-RSDW<15> bits, as will be discussed in more detail below. In response to the applied DRCLK0-DRCLK15 signals, the latches 440A-P store the applied RSDW<0>-RSDW<15> bits, respectively, and apply the stored bits through corresponding buffers 442A-P as the data bits D<0>-D<15> on the data bus DQ. The DRCLK0-DRCLK15 signals thus determine when each latch 440A-P places the corresponding RSDW<0>-RSDW<15> bit onto the corresponding line of the data bus DQ relative to transitions of the RCLK signal. This is true because each DRCLK0-DRCLK15 signal corresponds to one of the clock signals 438A-N selected by the corresponding multiplexer 436A-P responsive to the applied CMPDH0<0:3>-CMDPH15<0:3> word, and each clock signal 438A-N has a defined phase shift relative to the RCLK signal as previously described. The memory device 404 further includes a data clock driver circuit 446 that generates the data clock signals DCLK0 and DCLK1 in response to the read clock signal RCLK. Although the two data clocks DCLK0, DCLK1 are discussed in the described embodiments, only one data clock may be used in an alternative embodiment as will be understood by those skilled in the art.

During the read synchronization mode of operation, the read data pattern generator 424 generates successive 16-bit read synchronization data packet words RSDW<0:15>, with respective bits RSDW<0>-RSDW<15> in each word being clocked into respective data latches 440A-P in response the DCLK0-DCLK15 signals. During normal read operations, the 16-bit data packet words D<0:15> corresponding to the data being accessed are successively output from the multiplexer 122 (Figure 3) and are clocked into the data latches 440A-P responsive to the DCLK0-DCLK15 signals and thereafter applied through the buffers 442A-P and onto the data bus DQ. For ease of explanation and clarity of description, Figure 5 illustrates only the read data pattern generator 424 and corresponding RSDW<0:15> words being applied to the latches 440A-P. The driver

circuit 446 outputs the DCLK0, DCLK1 signals along with the RSDW<0:15> words being successively placed on the data bus DQ during the read synchronization mode and outputs the selected one of the DCLK0, DCLK1 signals along with the 16-bit data packet words being successively placed on the data bus DQ during the normal read mode. The read data pattern generator 424 may generate a variety of data patterns, and in one embodiment the read data pattern generator 424 generates a 15-bit repeating pseudo-random bit sequence as described in U.S. Patent Application No. 09/143,033 entitled METHOD AND APPARATUS FOR RESYNCHRONIZING A PLURALITY OF CLOCK SIGNALS USED TO LATCH RESPECTIVE DIGITAL SIGNALS, AND MEMORY DEVICE USING SAME, to Manning, which is incorporated herein by reference.

During the read synchronization mode of operation, the memory controller 402 sequentially latches data words D<0:15> applied on the data bus DQ by the memory device 404 in response to the selected DCLK0, DCLK1 signal. For each bit in the latched D<0:15> words, the memory controller 402 executes a synchronization process to determine whether the latched bit D<0>-D<15> was successfully captured in response to the selected one of the DCLK0, DCLK1 signals. For example, the controller can generate an expected value for the bit D<0>-D<15> and compare the latched bit to the expected value. When the two bits are equal, the controller 402 determines the bit was successfully captured, and otherwise determines the capture was unsuccessful. The controller 402 then sends a phase adjustment command to the memory device 404 over the CA bus. In response to the phase adjustment command, the device 404 adjusts the timing offset between the selected DCLK0, DCLK1 signal and the bit D<0:15> being synchronized, and thereafter once again determines whether the bit is successfully captured.

The overall operation of the read synchronization system 400 in synchronizing the data clock signals DCLK0 and DCLK1 during the read synchronization mode of operation will now be described in more detail. The memory controller 402 operates in the read synchronization mode in response to a predetermined condition. For example, the memory controller 402 may operate in the read synchronization mode of

operation as part of an initialization and synchronization procedure during which the memory controller 402 also synchronizes the command clock signal CCLK and the data clock signals DCLK0 and DCLK1 during write operations, and further performs other functions in initializing the memory device 404. Alternatively, the memory controller 402 may operate in the read synchronization mode of operation after a predetermined time during normal operation of the memory controller and memory device 404 in order to periodically resynchronize the data clock signals DCLK0 and DCLK1.

After commencing operation in the read synchronization mode of operation, the memory controller 402 applies phase adjustment commands to the memory device 404. Each phase adjustment command includes information identifying a particular memory device 404 and a particular bit D<0>-D<15> that is to be synchronized. contains information that is utilized by the memory device 404 to adjust the value of the phase command word CMDPH<0:3> stored in a corresponding phase command register 434A-P, as will be explained in more detail below. The phase adjustment command also selects one of the DCLK0, DCLK1 signals with which each D<0>-D<15> bit is to be synchronized. As previously mentioned, the value of the CMDPH<0:3> word adjusts the timing offset between the corresponding bit D<0>-D<15> and transitions of the selected data clock signal DCLK0 and DCLK1, and in this way the memory controller 402 utilizes the phase adjustment commands to adjust this offset for each bit D<0>-D<15>.

The command buffer and address capture circuit 46 latches the applied phase adjustment commands read in response to the ICLK signal as previously described with reference to Figure 3, and outputs the latched command to the command decoder and sequencer 408. The command decoder and sequencer 408 decodes the command portion of the applied phase adjustment command and thereafter generates the control signals 410 to place the memory device 404 in the read synchronization mode of operation. As part of placing the memory device 404 in the read synchronization mode, the up/down phase counter-controller 416 stores initial values for the CMDPH0-CMDPH15<0:15> words in the phase command registers 434A-P, and the read data pattern generator 424 begins

sequentially applying the read synchronization data words RSDW<0:15> to the data latches 440A-P. The command decoder and sequencer 408 also generates the phase adjustment command word PHADJCMD<0:Y> in response to the applied phase adjustment command from the controller 402.The value of the generated phase adjustment command word PHADJCMD<0:Y> is determined by the value of the phase adjustment command, and in this way the memory controller 402 controls the value of the phase adjustment command word PHADJCMD<0:Y>. As previously mentioned, the PHADJ<0:N> word contains information identifying a "selected" bit, which corresponds to the bit D<0>-D<15> on the data bus DQ that is being synchronized, and the PHADJCMD<0:Y> word similarly contains information identifying the selected bit.

The developed PHADJCMD<0:Y> word is applied to the counter-controller 416, which first examines the latched PHADJCMD<0:Y> word to determine the selected bit and thereafter reads the current value of the phase command word CMDPH<0:3> stored in the phase command register 434A-P associated with the selected bit. For example, if the PHADJCMD<0:Y> word identifies the bit D<0>, the counter-controller 416 reads the value of the CMDPH0<0:3> word stored in the phase command register 434A. After reading the value of CMDPH<0:3> word stored in the selected phase command register 434A-P, the counter-controller 416 generates a new value for the CMDPH<0:3> word responsive to the PHADJCMD<0:Y>. For example, the PHADJCMD<0:Y> word may contain information instructing the counter-controller 416 to increment or decrement the value of the read CMDPH<0:3> word. Once the counter-controller 416 has generated the new CMDPH<0:3> word, the counter-controller 416 stores the new word in the appropriate phase command register 434A-P. For example, if bit D<0> is being synchronized, the counter-controller 416 stores the newly generated value for the CMDPH0<0:3> word in the register 434A.

The counter-controller 416 can, alternatively, simply store an updated phase command word CMDPH<0:3> in the appropriate register 434A-P responsive to the PHADJCMD<0:Y> word. In this way, the counter-controller 416 need not first read the

CMDPH<0:3> word stored in the register 413A-P being updated. In another embodiment of the system 400, the counter-controller 416 can simultaneously adjust the values of the CMDPH0<0:3>-CMDPH15<0:3> words stored in the registers 434A-P, respectively. In this embodiment, the PHADJCMD<0:Y> word includes information the counter-controller 416 uses in independently adjusting each of the CMDPH0<0:3>-CMDPH15<0:3> words.

At this point, the read synchronization data words RSDW<0:15> generated by the pattern generator 424 are clocked into the data latches 440A-P responsive to the clock signals DRCLK0-DRCLK15, respectively, with each of these clock signals having an offset relative to the RCLK signal. As previously described, each multiplexer 436A-P outputs a selected one of the clock signals 438A-N as the corresponding DRCLK0-DRCLK15 signal, with the selected clock signal being determined by the value of the CMDPH0-CMDPH15<0:3> word applied to the multiplexer from the corresponding phase command register 434A-P. Thus, the data bits RSDW<0>-RSDW<15> are clocked out of the data latches 440A-P having respective timing offsets relative to the RCLK signal and thus relative to the DCLK0, DCLK1 signals, with each timing offset being determined by the value of the CMDPH0-CMDPH15<0:3> word store in the corresponding phase command register 434A-P.

At this point, in the memory controller 402 latches applied D<0:15> words responsive to the selected DCLK0, DCLK1 signal, and then determines whether the bit D<0>-D<15> being synchronized in the latched word was successfully captured. The memory controller 402 compares the latched value of the bit D<0>-D<15> being synchronized applied data word D<0:63> to the expected value for the bit and stores the result of the comparison. The Memory controller 402 thereafter generates a second phase adjustment command including an incremented phase value, and applies this new command to the memory device 404. The command buffer and address capture circuit 46 once again latches the applied phase adjustment command and outputs the latched command to the command decoder and sequencer 408. In response to the incremented phase value in this new phase adjustment command, the command decoder and sequencer 408 applies a new

PHADJCMD<0:Y> word to the counter-controller 416 which, in turn, reads the current value of the selected CMDPH0<0:3>-CMDPH15<0:3> word and increments or decrements the current value to develop a new CMDPH0<0:3>-CMDPH<15> word. The counter-controller 416 then stores the new value of the CMDPH0<0:3>-CMDPH<15> word in the corresponding register 434A-P. At this point, the RSDW<0>-RSDW<15> bits from the pattern generator 424 are clocked into the data latches 440A-P responsive to the clock signals DRCLK0-DRCLK15, respectively, with each of these clock signals having an offset relative to the DCLK0, DCLK1 signals as determined the current values of the CMDPH0<0:3>-CMDPH15<0:3> words. The RSDW<0:15> words are thus clocked out of the data latches 440A-P as the data initialization packet words D<0:15> on the data bus DQ, each bit D<0>-D<15> having a timing offset relative to the selected DCLK0, DCLK1 signal with the selected bit D<0>-D<15> currently being synchronized having a new timing offset as determined by the new value of the corresponding CMDPH0<0:3>-CMDPH<0:3> word.

The memory controller 402 once again captures from the data bus DQ the data initialization packet words D<0:15> having the new timing offset applied to the bit D<0>-D<15> being synchronized, compares the latched value to an expected value, and stores the result of this comparison. The memory controller 402 continues adjusting the value of the phase adjustment command and applying the adjusted commands to the memory device 404 in order to adjust the timing offset between the selected D<0>-D<15> bit and the selected data clock signal DCLK0, DCLK1. As the memory controller 402 adjusts the values of the phase adjustment commands, the memory controller stores a number of comparison results, each comparison result value corresponding to a particular value of the phase adjustment command (i.e., a particular timing offset of the selected bit D<0>-D<15> relative to the selected DCLK0 and DCLK1 signal). After a predetermined number of comparison results have been stored, the memory controller 402 executes a phase selection procedure to select a final phase adjustment command from among the phase adjustment commands that resulted in the successful capture of the selected bit

D<0>-D<15> as indicated by the corresponding comparison result. In one embodiment, the memory controller 402 stores sixteen comparison results, each corresponding to one of sixteen values for the phase adjustment command, and selects the final phase adjustment command from among the ones of the sixteen values that resulted in the successful capture of the selected D<0>-D<15> bit. One procedure that may be executed by the control circuit 406 in determining the final phase adjustment word PHADJ<0:4> is described in more detail in U.S. Patent No.5,953,284 to Baker et al., entitled METHOD AND APPARATUS FOR ADAPTIVELY ADJUSTING THE TIMING OF A CLOCK SIGNAL USED TO LATCH DIGITAL SIGNALS, AND MEMORY DEVICE USING SAME, which issued September 14, 1999 and which is incorporated herein by reference.

After selecting the desired one of the phase adjustment commands, the memory controller 402 applies the selected phase adjustment command to the memory device 404. The command buffer and address capture circuit 46 and command decoder and sequencer 408 in the memory device 404 then operate as previously described develop the selected PHADJCMD<0:Y> word corresponding to the selected phase adjustment command. The counter-controller 416 receives the selected PHADJCMD<0:Y> word, which includes information indicating that the word corresponds to the final selected value for the corresponding bit D<0>-D<15>. In response to the selected PHADJCMD<0:Y> word, the counter-controller 416 operates as previously described to update the value of the CMDPH0<0:3>-CMDPH<0:3> word stored in the register 434A-P corresponding to the bit being synchronized to a final value as determined by the selected PHADJCMD<0:Y> word. For example, if the D<0> bit is being synchronized, the counter-controller 416 sets the value of the CMDPH0<0:3> word stored in the register 434A to a final value as determined by the selected PHADJCMD<0:Y> word.

Upon the final phase command word CMDPH0<0:3>-CMDPH15<0:3> being stored in the corresponding register 434A-P, the corresponding clock signal DRCLK0-DRCLK15 signal has a phase relative to the read clock signal RCLK that is determined by the final phase command word, and this phase is utilized during normal read

operations of the memory device 404. This final phase command word CMDPH0<0:3>-CMDPH15<0:3> defines the timing offset between the selected DCLK0, DCLK1 signal and the corresponding bit D<0>-D<15>. This timing offset is then used during normal read operations of the memory device 404 so that the memory controller 402 may successfully capture this bit in the data packet words D<0:15> being transferred to the memory controller in response to read commands from the controller. The memory controller 402 control circuit 406 thereafter develops a data clock offset to edge align the second DCLK0, DCLK1 signal with the selected data clock signal. That enables the controller 402 to successfully capture data bits D<0>-D<15> responsive to either DCLK0, DCLK1 signal.

After synchronizing both DCLK0, DCLK1 signals for the selected bit D<0>-D<15>, the memory controller 402 develops a new phase adjustment command identifying the next bit D<0>-D<15> to be synchronized, and the memory controller 402 and memory device 404 thereafter operate in the same way as just described to synchronize the newly selected bit. The controller 402 and memory device 404 repeat this process for each of the bits D<0>-D<15> to independently synchronize each bit with the DCLK0, DCLK1 signals. When each bit D<0>-D<15> has been synchronized, the phase command registers 434A-P store final phase command values CMDPH0<0:3>-CMDPH15<0:3> to define the respective timing offsets between the each bit D<0>-D<15> and the DCLK0, DCLK1 signals. As will be understood by those skilled in the art, the exact process executed in synchronizing each D<0>-D<15> bit may be varied. For example each bit D<0>-D<15> may first be synchronized with the DCLK0 signal, and then each synchronized with the DCLK1 signal. Alternatively, a selected D<0>-D<15> bit may be synchronized with the DCLK0 signal, then the next selected bit synchronized with the DCLK0, signal, and so on sequentially for all bits. In another process, respective D<0>-D<15> bits may be independently synchronized with the DCLK0, signal in parallel. Other processes and combinations of the described processes may be utilized and will be well understood by those skilled in the art.

In another embodiment of the synchronization system 400 of Figure 5, the memory device 404 includes a shadow register 450, which is indicated with dotted lines, coupled between the counter-controller 416 and the phase command registers 434A-P. The shadow register 450 stores all the current CMDPH0-15<0:3> words, and functions as a storage "pipeline" between the counter-controller 416 and the registers 434A-P. In the system 400, the phase command registers 434A-P will typically be located near the data latches 440A-P and buffers 442A-P, which will all be physically proximate the external data bus DQ terminals of the memory device 404. In contrast, the counter-controller 416 may not be located physically near the registers 434A-P, thus causing the transfer of CMDPH0-15<0:3> words between the registers and the counter-controller to be slowed down due to the physical lengths of the data lines interconnecting these two components, as will be appreciated by those skilled in the art. If the counter-controller 416 must wait until an updated CMDPH0-15<0:3> word has been stored in the corresponding register 434A-P, the operation of the system 400 may be slowed down due to the delay in transferring the words between the counter-controller and the registers. With the shadow register 450, the counter-controller 416 can quickly update the value of one of the CMDPH0-15<0:3> words responsive to a corresponding phase adjustment command word PHADJCMD<0:Y> and thereafter begin processing a subsequent PHADJCMD<0:Y> word. The shadow register 450 thereafter transfers the updated CMDPH0-15<0:3> word to the proper register 434A-P while the counter-controller 416 is processing the subsequent command word PHADJCMD<0:Y>.

Figure 6 is a signal timing diagram that will be utilized to illustrate the relationship between the phase shift of the DRCLK0-DRCLK15 signals relative to the RCLK signal as defined by the respective CMDPH0<0:3>-CMDPH<15> words, and also illustrates the corresponding timing offset between the DCLK0 signal and respective bits D<0>-D<15> applied on the data bus DQ. In the example of Figure 5, the DCLK0 signal, which is generated by the drivers 446 in response to the RCLK signal, has the same phase as the RCLK signal. Figure 5 illustrates the three different delayed read clock signals

DRCLK0, DRCLK1, and DRCLK15 corresponding to values CMDPH0<0:3>, CMDPH1<0:3>, and CMDPH15<0:3> of the phase command word stored in the registers 434A, 434B, and 434P, respectively. As shown, the DRCLK0 signal has a phase shift ϕ_1 relative to the DCLK0 signal. When the DRCLK0 signal goes high at a time t_1, the corresponding data bit D<0> is placed on the data bus DQ have a timing offset t_{off1} relative to the rising edge of DCLK0 signal at a time t_0. As previously discussed, the data bit D<0> is offset by the time t_{off1} to enable the memory controller 402 to successfully latch the data bit in response to the DCLK0 signal.

In the second example of Figure 5, the DRCLK1 signal corresponding to the CMDPH1<0:3> word has a phase shift ϕ_2 relative to the DCLK0 signal, and the corresponding data bit D<1> has a timing offset t_{off2} relative to the rising edge of the DCLK0 signal at the time t_0. Thus, in this example the data bit D<1> is applied on the data bus DQ at a time t_2 before the rising edge of the DCLK0 signal at the time t_0. The third example illustrates the DRCLK15 signal having a phase shift ϕ_3 relative to the DCLK0 signal, and the corresponding data D<15> has a timing offset t_{off3} relative to the rising edge of the DCLK0 signal at the time t_0. In this example, the data D<15> is applied on the data bus DQ at a time t_3 before the rising edge of the DCLK0 signal at the time t_0. As illustrated by these examples, the phase command words CMDPH0<0:3>, CMDPH1<0:3>, and CMDPH15<0:3> are independently adjusted to thereby vary the phase shift of the DRCLK0-DRCLK15 signals relative to the DCLK0 signal. As the respective phase shifts of the DRCLK0-DRCLK15 signals are adjusted relative to the DCLK0 signal, the timing offsets of the respective data bits D<0>-D<15> in each data packet D<0:15> applied on the data bus DQ are adjusted relative to transitions of the DCLK0 signal by an amount corresponding to the adjusted phase shifts.

One skilled in the art will realize that the procedure executed by the control circuit 406 in synchronizing the data clock signals DCLK0, DCLK1 may vary. For example, in the above-described procedure the control circuit 406 captures only one data packet D<0:63> at each phase of the DRCLK0-DRCLK15 signal corresponding to the bit

D<0>-D<15> being synchronized. In another embodiment, the control circuit 406 may perform a predetermined number of comparisons at each given phase of the DRCLK0-DRCLK15 signal and timing offset of the corresponding D<0>-D<15> bit relative to the data clocks DCLK0 and DCLK1. In this embodiment, the control circuit 406 may, for example, control components of the memory controller 402 so that eight data packets D<0:63> are captured and compared at each phase of the DRCLK0-DRCLK15 signals. When all eight of these comparisons indicate successful captures, the control circuit 406 stores a "1" indicating successful data capture at this phase. However, if any of the comparisons at a given phase indicates an unsuccessful capture, the control circuit 406 stores a "0" indicating failure at this phase. Once again, after sixteen, for example, results signals have been stored, the control circuit 406 determines the final phase adjustment word PHADJ<0:4> and transfers this word to the memory device 404. The memory device 404 then operates as previously described, applying the corresponding PHADJCMD<0:Y> word to the counter-controller 416 which, in turn, stores the final phase command word CMDPH0<0:3>-CMDPH15<0:3> in the appropriate phase command register 434A-P to thereby set the final phase of the corresponding DRCLK0-DRCLK15 signal being adjusted.

The overall operation of the read synchronization system 400 and general operation of several components within that circuit have now been described with reference to Figure 4. More detailed circuitry for implementing the components of the read synchronization system 400 contained within the memory controller 402 and memory device 404 will be understood by those skilled in the art, and are not provided in more detail to avoid unnecessarily obscuring the present invention. Several of these components are described in more detail in U.S. Patent No. 6,029,250 to Keeth entitled METHOD AND APPARATUS FOR ADAPTIVELY ADJUSTING THE TIMING OFFSET BETWEEN A CLOCK SIGNAL AND DIGITAL SIGNALS TRANSMITTED COINCIDENT WITH THAT CLOCK SIGNAL, AND MEMORY DEVICE AND SYSTEM USING SAME, which is incorporated herein by reference. In addition, in addition to being used with a programmable delay circuit, such as the delay-locked loop 418, the present invention may

be used with other circuits for adjusting the timing offset between electrical signals, such as the command delay rings 506a described in U.S. Patent Application. No. 09/201,519 to Keeth, entitled METHOD AND APPARATUS FOR HIGH SPEED DATA CAPTURE UTILIZING BIT-TO-BIT TIMING CORRECTION, AND MEMORY DEVICE USING SAME, which was filed on November 30, 1998 and which is incorporated herein by reference. Furthermore, as previously mentioned, the synchronization system 400 may be used in a variety of different types of memory devices in addition to the SLDRAM devices described herein, such as in RAMBUS type memory devices and in double-data-rate synchronous dynamic random access memory devices ("DDR SDRAMs"). In DDR SDRAMs, the timing offsets of respective bits on the data bus DQ are adjusted with respect to a data strobe signal DQS, as will be appreciated by those skilled in the art.

Figure 7 is a functional block diagram illustrating a memory system 700 including memory devices 702, 704 that include read synchronization circuits 703, 705 according to another embodiment of the present invention. Each of the read synchronization circuits 703, 705 includes components (not shown) of the read synchronization system 400 of Figure 5, as will be discussed in more detail below. In the memory system 700, a clock generator 706 generates a system read clock signal RCLK that is applied to the memory devices 702, 704 and is applied to a memory controller 708. During a read operations, the memory device 702, 704 being accessed supplies read data on a data bus DQ and the memory controller 708 latches the read data in response to the RCLK signal. During write operations, the memory controller 708 supplies write data on the data bus DQ and applies a write clock signal WCLK to the memory devices 702, 704 which, in turn, latch the write data in response to the write clock signal. The memory controller 708 applies command and address information to the memory devices 702, 704 over a command bus CMD and also applies a command clock signal CCLK that the memory devices utilize to latch the applied command and address information. One skilled in the art will appreciate that the command bus CMD may be a multiplexed bus including both command and address information when the memory devices 702, 704 are packetized

type memory devices such as SLDRAMs. Alternatively, the command bus CMD may include separate command and address busses when the memory devices 702, 704 are non packetized type memory devices such as a DDR SDRAM.

During normal write operations, the memory controller 708 applies a write command on the CMD bus to the memory devices 702, 704 which, in turn, latch the write command in response to the CCLK signal also applied by the memory controller. Each of the memory devices 702, 704 decodes the latched command and determines whether it is the device being accessed. The memory controller 708 supplies write data on the DQ bus and the memory device 702, 704 being accessed latches the supplied write data in response to the CCLK signal from the memory controller. During normal read operations, the memory controller 708 applies the read command on the CMD bus to the memory devices 702, 704 which, once again, latch and decode the read command. The memory device 702, 704 being accessed thereafter supplies read data on the DQ bus and the memory controller 708 latches the read data in response to the RCLK the signal from the clock generator 706. Each of the data bits on the DQ bus has a respective timing offset relative to the RCLK signal, with the timing offsets being determined during a read synchronization mode of operation. The respective timing offsets allow the memory controller 708 to successfully capture all read data bits supplied on the DQ bus during normal read operations. During the read synchronization mode of operation, the circuits 703, 705 operate in an analogous manner to components of the read synchronization system 400 of Figure 5. Such operation will be well understood by those skilled in the art in view of the previous detailed description of the read synchronization system 400 of Figure 5, and thus, for the sake of brevity, will not be described in more detail. Briefly, for each data bit on the data bus DQ a phase command CMDPH is stored in corresponding phase command register 434A-434P (see Figure 5) to define the timing offset between the data bit and the RCLK signal.

It is to be understood that even though various embodiments and advantages of the present invention have been set forth in the foregoing description, the above disclosure is illustrative only, and changes may be made in detail, and yet remain within the

broad principles of the invention. For example, many of the components described above may be implemented using either digital or analog circuitry, or a combination of both, and also, where appropriate, may be realized through software executing on suitable processing circuitry. Therefore, the present invention is to be limited only by the appended claims.